CLEARINGHOUSE FOR FEDERAL SCIENTIFIC AND TECHNICAL INFORMATION CFSTI
DOCUMENT MANAGEMENT BRANCH 410.11

LIMITATIONS IN REPRODUCTION QUALITY

ACCESSION # AD 604 284

- [x] 1. WE REGRET THAT LEGIBILITY OF THIS DOCUMENT IS IN PART UNSATISFACTORY. REPRODUCTION HAS BEEN MADE FROM BEST AVAILABLE COPY.
- [x] 2. A PORTION OF THE ORIGINAL DOCUMENT CONTAINS FINE DETAIL WHICH MAY MAKE READING OF PHOTOCOPY DIFFICULT.
- [ ] 3. THE ORIGINAL DOCUMENT CONTAINS COLOR, BUT DISTRIBUTION COPIES ARE AVAILABLE IN BLACK-AND-WHITE REPRODUCTION ONLY.
- [ ] 4. THE INITIAL DISTRIBUTION COPIES CONTAIN COLOR WHICH WILL BE SHOWN IN BLACK-AND-WHITE WHEN IT IS NECESSARY TO REPRINT.
- [ ] 5. LIMITED SUPPLY ON HAND: WHEN EXHAUSTED, DOCUMENT WILL BE AVAILABLE IN MICROFICHE ONLY.
- [ ] 6. LIMITED SUPPLY ON HAND: WHEN EXHAUSTED DOCUMENT WILL NOT BE AVAILABLE.
- [ ] 7. DOCUMENT IS AVAILABLE IN MICROFICHE ONLY.
- [ ] 8. DOCUMENT AVAILABLE ON LOAN FROM CFSTI (TT DOCUMENTS ONLY).
- [ ] 9.

NBS 9/64

PROCESSOR: Gr

604284

SOME COMBINATORIAL PROBLEMS ARISING IN THE THEORY OF MULTI-STAGE PROCESSES

Richard Bellman
Oliver Gross

P-456

13 November 1953

The RAND Corporation
1700 MAIN ST. • SANTA MONICA • CALIFORNIA

15

Summary: Using the technique of continuous approximation, approximate solutions to a number of important multi-stage scheduling problems are determined. In addition, the functional equation approach of the theory of dynamic programming is used to derive an alternate proof of a result of S. Johnson, contained in P-402.

Some Combinatorial Problems Arising in the Theory of Multi-Stage Processes

By

Richard Bellman and Oliver Gross

1. Introduction.

A problem of some importance in industrial applications which gives rise to some interesting and quite difficult combinatorial problems is the following:

"There are n items, not all identical, which have to be processed through a number of machines of different type. The order in which the machines are to be used is not immaterial, since some processes must be carried out before others. Given the times required by the i-th item on the j-th machine, a_{ij}, $i = 1,2,\ldots,n$, $j = 1,2,\ldots,m$, determine the order in which the items should be fed into the machines so as to minimize the total time required to complete the lot."

As a simple example of the above, consider the case where we have a number of books which must be printed and then bound. Clearly the printing must precede the binding.

Mathematically, the problem is one of arrangements, which can be solved in any particular case by enumeration. However, a quick count of the possible arrangements will show that as soon as the number of items reaches ten, the enumerative technique becomes unwieldy.

At the suggestion of one of the authors, this problem was investigated by S. Johnson, who found a number of interesting and important results which we shall describe below. He was able to solve completely the two-stage problem for any number of dissimilar items. The three

stage problem, however, poses difficulties of an entirely new and formidable type, and has so far resisted solution.

An important tool in Johnson's work was an explicit formula for idle time on the second machine in the two stage process. It is this quantity which determines the efficiency of a particular routing order.

In this paper we shall tackle the simplified problem of determining the optimal order when there are a large number of items of only a few different types. This is perhaps the most important problem as far as practice is concerned.

Even here, the original problem seems difficult to resolve. Consequently we shall use a device which works uniformly well throughout the theory of dynamic programming, namely the replacement of a discrete problem by a continuous version. As is frequently the case, the continuous version may be solved with great ease and elegance.

For those interested in other problems involving multi-stage processes which may be approximated by continuous processes and thereby resolved, we refer to [1], [4] and [5], where other references to the theory of dynamic programming may be found.

Finally we shall show how the two-stage process may be attacked by the functional equation approach of the theory of dynamic programming, cf [2], and resolved without the use of an explicit formula for the idle time. This method is important since it is not always possible to obtain a tractable explicit analytic representation of the quantity that is to be minimized or maximized in many analogous problems.

The methods presented here may also be used to to treat the case where there are interchangeable machines, operators trained to work some or all of the machines, and so on. We feel that it is best to expose the method in its native simplicity and leave the extensions to those to whom the problem may have some immediate interest.

2. The Results of Johnson.

In this section we shall present the results already obtained by Johnson. He first of all established by a simple argument that for the cases of two and three machines the order of processing on each machine can be taken to be the same in an optimal arrangement. This results in a considerable simplification and permits an explicit formula for the idle time to be derived. [4].

Let us note in passing that Johnson showed by means of a simple example that this identical ordering may not be valid for four or more stages.

Theorem 1 (Johnson). Let x_i be the inactive time on the second machine immediately before the i-th item is processed on the second machine. Let (a_i, b_i) be the times required to process the i-th item on the first and second machines respectively, and assume that the items are arranged in numerical order. Then

$$(1) \qquad I_{2n} = \sum_{i=1}^{n} x_i = \underset{1 \le u \le n}{\mathrm{Max}} \left[\sum_{i=1}^{u} a_i - \sum_{i=1}^{u-1} b_i \right]$$

where I_{2n} represents the total idle time on the second machine.

We shall give the proof for the sake of completeness, since it is quite short.

We have

$$x_1 = a_1 \tag{2}$$
$$x_2 = \max(a_1 + a_2 - b_1 - x_1, 0)$$

whence

$$x_1 + x_2 = \max(a_1 + a_2 - b_1, a_1) . \tag{3}$$

Similarly

$$x_3 = \max\left(\sum_{i=1}^{3} a_i - \sum_{i=1}^{2} b_i - \sum_{i=1}^{2} x_i, 0\right) \tag{4}$$

and

$$\sum_{i=1}^{3} x_i = \max\left(\sum_{i=1}^{3} a_i - \sum_{i=1}^{2} b_i, \sum_{i=1}^{2} x_i\right) \tag{5}$$
$$= \max\left(\sum_{i=1}^{3} a_i - \sum_{i=1}^{2} b_i, \sum_{i=1}^{2} a_i - b_1, a_1\right)$$

The remainder of the proof is inductive.

The two machine problem is equivalent to determining the arrangement of n items which will minimize the right side of (1). The solution is given by

Theorem 2(Johnson). <u>An optimal ordering is determined by the following rule: item i precedes item j if</u>

$$\min(a_i, b_j) < \min(a_j, b_i). \tag{6}$$

<u>If there is equality, either ordering is optimal, provided that it is consistent with all the definite preferences.</u>

For the three-stage problem, the corresponding formula for the total idle time on the third machine, also derived by Johnson, is

$$(7) \qquad I_{3n} = \max_{1 \le u \le v \le n} \left[\sum_{i=1}^{u} a_i - \sum_{i=1}^{u-1} b_i + \sum_{i=1}^{v} b_i - \sum_{i=1}^{v-1} c_i \right]$$

where a_i, b_i, c_i, denote respectively the times required by the ith item on the first, second and third machines.

3. Continuous Versions - 2 machines.

To illustrate our techniques, let us begin by considering the simplest case, that of two machines and two different types of items, assuming that the total number of items is large compared to the times required to process any individual item. In place of the expression $\sum_{i=1}^{u} a_i - \sum_{i=1}^{u-1} b_i$, we consider the integral

$$(1) \qquad I(u) = \int_0^u (a(t) - b(t))dt.$$

The analogue of an arrangement of items is a characteristic function ϕ defined over the interval $[0,T]$. This function determines $a(t)$ and $b(t)$ in the following way

$$(2) \qquad \begin{aligned} a(t) &= a_1\phi + a_2(1-\phi) \\ b(t) &= b_1\phi + b_2(1-\phi) \end{aligned}$$

where (a_1, b_1) and (a_2, b_2) represent the times required on the first and second machines for the first and second types respectively. The function $\phi(t)$ is the characteristic function of the set over which the first item is processed, and $1-\phi$ is the characteristic function of the set over the second item is processed. The constraints upon ϕ are that it take

on only the values 0 and 1, and in addition satisfy

$$\int_0^T \phi \, dt = k, \tag{3}$$

which is equivalent to the statement that k of the T items belong to the first type. If we set

$$\begin{aligned} \alpha &= (a_1 - a_2 + b_2 - b_1) \\ \beta &= (a_2 - b_2) \end{aligned} \tag{4}$$

the problem is that of determining the quantity

$$I = \underset{\phi}{\text{Min}} \underset{0 \le u \le T}{\text{Max}} \left[\alpha \int_0^u \phi \, dt + \beta \right], \tag{5}$$

and determining the corresponding functions ϕ .

It is to be expected that in general to obtain a minimum we must extend our class of functions and consider those which correspond to "mixed" policies. This is to say we must allow ϕ to satisfy the weaker condition $0 \le \phi \le 1$, rather than be restricted to values of 0 or 1. For two machines, there is a solution in the narrower class. For three machines, this will not be true in the majority of cases. This illustrates the complexity of the solution of the corresponding discrete problems.

Let us now demonstrate

<u>Theorem</u> 3. <u>A minimizing</u> ϕ <u>is given by</u>

(6) <u>If</u> $\alpha > 0$, $\phi = 1$ <u>in</u> $[T-k, T]$

$\phi = 0$ <u>in</u> $[0, T-k]$

<u>If</u> $\alpha < 0$, $\phi = 1$ <u>in</u> $[0, k]$

$= 0$ <u>in</u> $[k, T-k]$

<u>If</u> $\alpha = 0$, ϕ <u>is arbitrary</u>.

Proof: Let us consider $\alpha > 0$ first. Denote the ϕ described above by ϕ^P. Then for any ϕ satisfying (3) and $0 \le \phi \le 1$, it is clear that

(7) $$\alpha \int_0^u \phi \, dt + \beta u \ge \alpha \int_0^u \phi^P dt + \beta u$$

Hence

(8) $$\max_{0 \le u \le T} \left[\alpha \int_0^u \phi \, dt + \beta u \right] \ge \max_{0 \le u \le T} \left[\alpha \int_0^u \phi^P dt + \beta u \right]$$

which shows that ϕ^P furnishes the minimum. The case where $\alpha < 0$ is treated similarly. It is clear that for $\alpha = 0$, ϕ plays no role. This completes the proof.

The ϕ we have found is not necessarily unique. It is, however, the simplest solution, and the most important in applications.

Let us now consider the error made in using the solution to the continuous case as an approximation the solution of the discrete case. We see that the difference between $\sum_{i=1}^{u} a_i - \sum_{i=1}^{u-1} b_i$ and $\sum_{i=1}^{u} a_i - \sum_{i=1}^{u} b_i$ is b_1 or b_2, hence negligible if n is large. Now consider the difference between

(9) $$I(u) = \sum_{i=1}^{u} a_i - \sum_{i=1}^{u} b_i = \sum_{i=1}^{u} \left[(a_1 - b_1)\phi_i + (a_2 - b_2)(1 - \phi_i) \right] = \alpha \sum_{i=1}^{u} \phi_i + \beta u$$

$u = 1, 2, \ldots, n$, where ϕ_i is 1 if the first item occupies the i-th position and zero otherwise, and the continuous analogue, $J(u) = \alpha \int_0^u \phi \, dt + \beta u$. It is clearly sufficient to consider the difference at integral values of u. We have

(10) $$J(u) - I(u) = \alpha \left(\int_0^u \phi \, dt - \sum_{i=1}^{u} \phi_i \right).$$

For each ϕ_i consider the corresponding $\phi(t)$ which has the property that $\int_0^u \phi \, dt = \sum_{i=1}^{u} \phi_i$, when $u = 1, 2, \ldots, n$. The difference will then be zero.

It follows then that if we minimize $J(u)$ over all ϕ and find that the minimizing ϕ, or a minimizing ϕ, corresponds to a ϕ_i, we obtain a very good approximation to the solution of the original problem. Even, as below, in the case where there is no minimizing function corresponding to a characteristic function, we may still be able to approximate by means of characteristic functions, and obtain a useful answer. For example, if $\phi = a$, where $0 < a < 1$, in the interval $(0,s)$, we may consider an approximate ψ given by $\psi = 1$ in $(0,a)$ and 0 in (a,s). Still finer subdivisions will yield more accurate approximation. For a discussion of this concept of replacing mixed strategies by pure strategies, see [3]; for an application see [5].

The case where there are more than two types of items is treated in a similar manner. Take, for instance, the case of three types of items. Let ϕ_1, ϕ_2, ϕ_3 be the respective characteristic functions of the sets where each is processed. Then, corresponding to (5) we have the problem of determining the minimum of

$$(11) \qquad I(\phi) = \max_{0 \le u \le T} \left[\int_0^u \left[(a_1 - b_1)\phi_1 + (a_2 - b_2)\phi_2 + (a_3 - b_3)\phi_3 \right] dt \right.$$

over all ϕ_1, ϕ_2 and ϕ_3 satisfying the constraints,

$$(12) \qquad \begin{aligned} &(a) \quad 0 \le \phi_i \le 1 \\ &(b) \quad \phi_1 + \phi_2 + \phi_3 = 1 \\ &(c) \quad \int_0^T \phi_1\, dt = k_1, \quad \int_0^T \phi_2\, dt = k_2, \quad \int_0^T \phi_3\, dt = k_3, \\ &\qquad k_1 + k_2 + k_3 = T \end{aligned}$$

Using the result of (7), it is clear that the order of the items is determined by the quantities $a_i - b_i = d_i$, which are to be arranged in increasing order of magintude. This yields the optimal ordering.

4. Continuous Version. Three Machines.

Let us now discuss the three-machine problem assuming two distinct varieties of items. The method given here extends to any number of stages and any number of items, provided that we assume that the ordering on each machine is the same.

As the continuous analogue of the expression given for I_{3n} in (2.7) we have

$$(1)\qquad I_{(T)} = \max_{0 \le u \le v \le T} \left[\int_0^u (a(t)-b(t))dt + \int_0^v (b(t)-c(t))dt \right],$$

$$= \max_{0 \le u \le v \le T} \left[\alpha \int_0^u \phi \, dt + \beta u + \gamma \int_0^v \phi \, dt + \delta v \right]$$

upon setting

$$(2)\qquad a(t) = a_1\phi + a_2(1-\phi),\quad b(t) = b_1\phi + b_2(1-\phi),\quad c(t) = c_1\phi + c_2(1-\phi),$$

and

$$(3)\qquad \alpha = a_1 - a_2 + b_2 - b_1,\quad \beta = a_2 - b_2,\quad \gamma = b_1 - b_2 + c_2 - c_1,\quad \delta = b_2 - c_2.$$

We wish to determine the minimum of $I_{(T)}$ for all ϕ subject to

$$(4)\qquad 0 \le \phi \le 1,\quad \int_0^T \phi \, dt = k.$$

A solution is given by

<u>Theorem 4. The minimum value of I is</u>

$$(5)\qquad V(k,T) = \max\left(0,\ \gamma k + \delta T,\ (\alpha+\beta)k + (\gamma+\delta)T \right).$$

<u>A minimizing ϕ is given by</u>

$$(6)\qquad \phi^*(t) = k/T$$

<u>for $0 \le t \le T$. In general, the solution is non-unique.</u>

Proof: We have

(7) $$I_T(\phi) \geq 0, \qquad \text{setting } u = v = 0.$$
$$\geq \gamma k + \delta T, \qquad \text{setting } u = 0, v = T.$$
$$\geq (\alpha+\beta)k + (\gamma+\delta)T, \qquad \text{setting } u = T, v = T.$$

On the other hand, if we take $\phi = k/T$, we wish to determine the maximum of the linear form

(8) $$L(u,v) = \frac{\alpha k u}{T} + \beta u + \frac{\gamma k v}{T} + \delta v$$

over the region $0 \leq u \leq v \leq T$. The maximum occurs at the of the vertices $(0,0)$, $(0,T)$, or (T,T), and hence is as given in (5).

The inequalities of (7) show that $\operatorname{Min}_{\phi} I \geq V$. The argument above shows that $\operatorname{Min}_{\phi} \phi \leq V$. Hence we have equality.

If, for example, $0 \neq \gamma k + \delta T \neq (\alpha+\beta)k + (\gamma+\delta)T \neq 0$, the solution is non-unique. To show this, let us observe that the above condition implies that the values of the linear form at the vertices are such that one lies strictly above the other two. It follows that if we perturb ϕ^* by a sufficiently small amount over an interior interval in such a way as to leave its total integral over $(0,T)$ unchanged, the effect on L will be to leave its values at the vertices unchanged. Hence we can obtain arbitrarily many solutions from the original solution ϕ^*.

It is also possible to derive various conditions of quite special nature which will ensure uniqueness.

Incidentally, the foregoing minimization problem has a determinate two-person zero sum game interpretation in which the minimizing player picks the ϕ (subject to the same constraints as above), and the maximizing player picks a point (u,v) in the triangle $0 \le u \le v \le T$, with the payoff given by $\alpha \int_0^u \phi\, dt + \beta u + \gamma \int_0^v \phi\, dt + \delta v$. The value of the game is given by (5), an optimal strategy for the minimizing player is given by (6), while the maximizing player has an optimal pure strategy consisting of choosing the vertex which yields the largest value for the linear form, L.

5. Alternative Derivation of the Decision Function.

Let us in this section show how the Johnson criterion, given in Theorem 2, may by derived by use of the functional equation approach of the theory of dynamic programming.

Let

(1) $f(a_1, b_1, a_2, b_2, \ldots, a_N, b_N; t)$ = time consumed processing the N items with required times a_i, b_i on the first and second machines when the second machine is committed for t hours ahead, and an optimal scheduling procedure is employed.

Then if the first item is processed first, we have

(2) $f(a_1, b_1, \ldots, a_N, b_N; t) = a_1 + f(0, 0, a_2, b_2, \ldots, a_N, b_N; b_1 + \mathrm{Max}(t-a_1, 0))$.

If we choose the second item to follow, we obtain

(3) $$f(a_1, b_1, \ldots, a_N, b_N; t) = a_1 + a_2 + f(0, 0, 0, 0, a_3, b_3, \ldots, a_N, b_N; b_2 + \mathrm{Max}\,[b_1 - a_2 + \mathrm{Max}(t-a_1, 0), 0]).$$

On the other hand, if we interchange the orders, we obtain

$$(4)\quad f(a_1, b_1, \dots, a_N, b_N; t) = a_2 + a_1 + f(0, 0, 0, 0, a_3, b_3, \dots, a_N, b_N; b_1 + \mathrm{Max}\left[b_2 - a_2 + \mathrm{Max}(t - a_2, 0), 0\right]).$$

It follows from these formulae that the order of the items which minimizes the new t-term is optimal. It is not immediately obvious that this order is independent of t. We have

$$(5)\quad b_2 + \mathrm{Max}\left[b_1 - a_2 + \mathrm{Max}(t - a_1, 0), 0\right] = \mathrm{Max}\left[\mathrm{Max}(t - a_1, 0), a_2 - b_1\right] + b_1 + b_2 - a_2$$

$$= \mathrm{Max}\left[t - a_1, 0, a_2 - b_1\right] + b_1 + b_2 - a_2$$

$$= \mathrm{Max}\left[t, a_1, a_1 + a_2 - b_1\right] + b_1 + b_2 - a_1 - a_2$$

It is easily seen from this that regardless of the value of t, the interchange which minimizes Max $\left[a_1, a_1 + a_2 - b_1\right]$ cannot increase the total time, and may decrease it.

The remainder of the proof proceeds as in Johnson's paper.

BIBLIOGRAPHY

1. Bellman, R. "Bottleneck Problems and the Theory of Dynamic Programming," Proc. Nat. Acad. Sci., 39 (1953), p. 947.

2. ———. An Introduction to the Theory of Dynamic Programming. RAND Corporation Report No. 245, 1953.

3. Bellman, R. and D. Blackwell. "On Games Involving Bluffing," Proc. Nat. Acad. Sci., 35 (1949), pp. 600-605.

4. Bellman, R., I. Glicksberg, and O. Gross. "Some Variational Problems Occurring in the Theory of Dynamic Programming," Proc. Nat. Acad. Sci., 39 (1953), pp. 298-301.

5. Bellman, R. and S. Lehman, "The Continuous Gold-mining Equation," Proc. Nat. Acad. Sci. (to appear).

6. Johnson, S. "Optimal Two- and Three-stage Production Schedules with Setup Times Included," RAND Corporation Paper No. 402, 1953.